full circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU COMMUNITY

ubuntu
kubuntu
xubuntu
edubuntu

SPREADING UBUNTU

ONE MACHINE AT A TIME

INSTALL ANY DISTRO TO YOUR ASUS EEE PC

full circle

www.fullcirclemagazine.org

WIN A COPY!
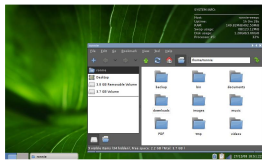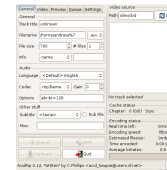
P.08

P.14

P.23

P.24

P.17

P.19

P.28

P.33

icons: KDE4 Oxygen

EDITORIAL

Welcome to another issue of Full Circle Magazine.

t's that time again folks. With Jaunty being released in a few months, a new codename is released. See our news item about 9.10 on .

Our Web and C series are joined this month with *Spreading Ubuntu*. In this series of articles, Grant Paton-Simpson will give you advice on which computer users you could suggest Ubuntu to. We also have another series in the making which will start in a few months time. Keep your eyes peeled! Also starting this month is *Modern Times*, a cartoon by Costantinos Bourboulas, it's on .

Returning this month is the book review, with competition. This month it's *Ubuntu For Non-Geeks 3rd Edition*. You can read my review on . Hopefuly in the next few months we'll have several copies of the reviewed book as prizes.

Good luck in the competition, and keep those emails coming!

All the best,
Ronnie
Editor, Full Circle Magazine
ronnie@fullcirclemagazine.org

This magazine was created using :

What is Ubuntu?

Ubuntu is a complete operating system that is perfect for laptops, desktops and servers. Whether at home, school or work Ubuntu contains all the applications you'll ever need including word processor, email application and web browser.

Ubuntu is and always will be free of charge. You do not pay any licensing fees. You can download, use and share Ubuntu with your friends, family, school or business for absolutely nothing.

Once installed, your system is ready to use with a full set of productivity, internet, drawing and graphics applications, and games.

http://url.fullcirclemagazine.org/7e8944

Washington Times Goes Open Source

The Washington Times has announced that it uses Ubuntu on its servers and is working on several Python-based projects:

The Washington Times has used open source projects for some time now. All of our servers run Ubuntu linux. Our database servers use PostgreSQL. Our web servers use lighttpd and Apache httpd, and memcached. We develop entirely in python, using the django framework. We are obviously strong believers in open source.

http://opensource.washingtontimes.com/

Ubuntu 9.10: Karmic Koala

Mark Shuttleworth has announced the successor to Ubuntu 9.04 : Jaunty Jackalope

*Ladies and gentlemen, allow me to introduce the **Karmic Koala**, the newest member of our alliterative menagerie.*

When you are looking for inspiration beyond the looming Jaunty feature freeze, I hope you'll think of the Koala, our official mascot for Ubuntu 9.10. And if you'll bear with me for a minute I'll set the scene for what we hope to achieve in that time.

For the 9.10 desktop:

First impressions count. We're eagerly following the development of kernel mode setting, which promises a smooth and flicker-free startup. We'll consider options like Red Hat's Plymouth, for graphical boot on all the cards that support it. We made a splash years ago with Usplash, but it's time to move to something newer and shinier. So the good news is, boot will be beautiful. The bad news is, you won't have long to appreciate it! It only takes 35 days to make a whole Koala, so we think it should be possible to bring up a stylish desktop much faster.

The desktop will have a designer's fingerprints all over it - we're now beginning the serious push to a new look. Brown has served us well but the Koala is considering other options. Come to UDS for a preview of the whole new look.

You can read the full announcement at: http://fridge.ubuntu.com/

The Adventures of Rick Rocket released for Linux
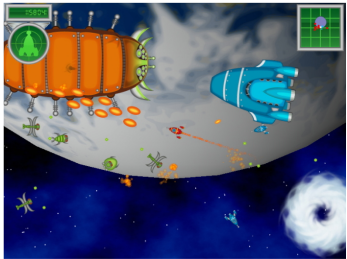
The full version features 48 missions, epic space battles set in a wide variety of locations, over 30 ships that show damage as they are hit, 12 powerups, colorful cartoon graphics, a superb music soundtrack, and more!

A trial version with 8 free missions can be downloaded from:

http://www.mygamecompany.com/Products/RickRocket/main.htm

The full version of Rick Rocket costs $19.99 (USD).

For more Linux game news, jump to page 28 for **Ubuntu Games**.

Ballmer: Linux Bigger Competitor than Apple
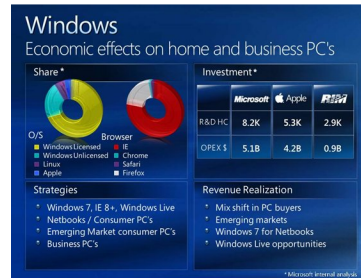
Microsoft's CEO had some interesting things to say about which companies Microsoft sees as its competitors in the client operating system space. You'd think Apple was their number one competitor - and you'd be wrong. According to Microsoft, Linux is a bigger threat to the company than Apple, placing Linux above Apple in the marketshare figure pie chart thing. "Linux, you could see on the slide, and Apple has certainly increased its share somewhat," Ballmer said.

In any case, it appears that Linux (and piracy) is a larger blip on Microsoft's radar than Apple, and it's not hard to see why. With an economy that's not doing very well, people will opt for cheaper products. Apple cannot offer those, but Linux and piracy can.

http://www.osnews.com

HP To Certify Ubuntu On Proliant Servers

HP and Canonical are working together to achieve full Hewlett-Packard certification of Ubuntu on Proliant servers.

Mark Murphy, alliances manager for Canonical, said the goal of the partnership is for HP to list Ubuntu as a supported operating system.

"Furthermore, both companies are fully cooperating at the engineering level to provide full underlying confidence for HP customers using the certified servers," Murphy wrote in a blog post.

Ubuntu has recently been making great strides on the desktop. Last October, Canonical's release of Ubuntu 8.10 Desktop Edition, nicknamed "Intrepid Ibex," was praised by many testers -- including CRN's Test Center -- for its robust functionality and ease of use.

http://www.crn.com/

Ever had a video clip you wanted to reduce in size and add to a slideshow for a presentation? Or, maybe convert into another format so that you could play it? Today, I will show you how to do both of those things using the command-line tool 'ffmpeg'. I will also introduce you to the 'mogrify' command, which is an image-editing tool contained in the imagemagick package. It allows you to do many things, but I will cover the basics – mostly the resizing of images (e.g. for thumbnails or other small images).

To use these tools, you will need to install ffmpeg and imagemagick through either Synaptic Package Manager or Add/Remove Applications, or, in the spirit of this article, through the command-line with:

```
sudo apt-get install ffmpeg
imagemagick
```

It's safe to run the install command – if you're unsure whether or not you've already installed it – since apt-get will not overwrite the existing program, but merely inform you that it's already installed. Also, it will ask you for your password (since you're using sudo), and, if you're new to this, you may be surprised that it does not show anything when you enter your password. This is normal, just type your password and hit the enter key.

For this article I will be converting a short clip from "Freedom Downtime" which I used in a presentation. ffmpeg offers a lot of options (which you can read about in detail in the roughly 13-page-long manpage – by using the command "man ffmpeg"), but the option I use most frequently is the option to convert files. The format for such a command is:

```
ffmpeg -i inputfile.filetype
outputfile.filetype
```

That command would just convert the input file to whatever output file you specified in 'output.filetype' – without changing the size (since, if the size is not specified, ffmpeg defaults to the size of the source). However, to convert "Freedom Downtime" (starting size of 640x480) to a smaller clip (say 320x240), the command would look like this:

```
ffmpeg -i freedom\
downtime.mpg -s 320x240
freedom\ downtime\ resized.mpg
```

Of course, you could also change the format of the file at the same time by changing the file extension. This shouldn't take too long (obviously depending on the size of the clip). My clip was a few minutes long, and took maybe 30 seconds to convert on my laptop.

Mogrify is a very handy tool, especially if you find yourself posting a lot of pictures on web forums that don't allow uploads of images exceeding a certain size, or do not allow linking to large images for previews in posts. I use mogrify mainly for making thumbnails of images, but it can do many more things,

such as adding text, adding effects (charcoal, colorized, etc.) and much more (it's all outlined, once again, in the manpage, reached with "man mogrify"). Mogrify supports resize arguments in either percentages or pixel values. So, for example, if you have a 1280x800 pixel image which you want to resize to 640x400, you can do so with either:

```
mogrify –resize 50 in.jpg
out.jpg
```

or:

```
mogrify –resize 640x400 in.jpg
out.jpg
```

or even just:

```
mogrify –resize 50% in.jpg
out.jpg
```

However, if the pixel measurements and the aspect-ratio you supply are different, it could result in a smaller image than expected, as it will scale to the nearest values that are still in proportion. Mogrify also has a thumbnail argument, which does roughly the same as the resize option, yet it removes any unnecessary comments, etc. from the file header to reduce image size. So using that tool the command would be:

```
mogrify –thumbnail 50 in.jpg
out.jpg
```

You can also use it to convert images using the "-format" argument. So,

```
mogrify –format jpg *.png
```

will convert all the .png files in the current directory into .jpg format (names will be the same).

As you can see, contrary to common belief, command-line tools can help with graphical projects, and often do it more quickly or more effectively than a GUI program with confusing menus and/or different layouts in newer versions. The command will (usually) stay the same, and arguments are very rarely changed. And so, command-line tools are much more universal – which is why users on ubuntuforums.org usually supply commands instead of GUI methods for solutions, since the commands apply to Kubuntu, Xubuntu and Ubuntu, as well as other systems too. Hopefully, you've found this useful, and next time you need to convert a video or an image, you'll remember mogrify and ffmpeg. After all, practice makes perfect.

Further Reading:

http://www.imagemagick.org/www/mogrify.html – Very useful guide on imagemagick in general, from the imagemagick site

http://www.ffmpeg.org/documentation.html - The official ffmpeg documentation with useful information, FAQs, etc.

Lucas has learned all he knows from repeatedly breaking his system, then having no other option but to discover how to fix it. When he finds time, he also publishes a blog at http://lswest-ubuntu.blogspot.com.

U ntil now, I have been presenting to you some code, and instructions on how to compile and execute that code. So, until now, you've probably only needed an editor (emacs, vi, ...) and a compiler suite (gcc). However, there are some other tools which make developing code easy (bear in mind that developing doesn't mean only typing source code; it covers the entire process - typing code, compiling code, testing code, etc.), and there are many tools out there. There are even some IDEs (integrated development environments) which combine some of these tools in a nice graphical front-end (e.g. the CDT in the Eclipse project, kdevelop, Code::blocks, anjuta, and more), but it is my humble opinion that a beginning programmer should first get an idea about how things work under the hood, before he starts using shortcuts. Although there are many tools out there which cover many categories, in this article we will focus on troubleshooting code/applications.

strace and ltrace

strace is one of my best friends. ltrace is also a nice tool, but I don't make use of it that often. You can install both by typing

```
sudo apt-get install strace ltrace
```

Now, what do they do? strace intercepts the system calls made by processes. A system call is a function call which transfers control to the kernel for some functions that can be done in user space. For example, incrementing a variable is translated to a simple assembler command, but when you do something which deals with resources, this always goes past your kernel. If you read 'man 2 syscalls', you can get a list of all system calls the kernel supports. Now, why is this a good thing to monitor? Well, if you monitor which system calls your application makes, you can follow the logical flow of your application, and the good thing is that it is non-intrusive, which means you can run it on every binary on your system. As an example, I will highlight the output of wget, installable by typing:

```
sudo apt-get install wget
```

wget is an application which

fetches an URL from the Internet and writes it to disk.

If we take a look at the output of

```
strace wget -q
http://www.google.com
```

shown in Fig. 1 (right) we can clearly see some interesting parts in the execution.

We can see that it starts with a call to 'execve()' (you can use man execve; this goes for each systemcall - the first word on a line echoed by strace), which loads the binary. A bit later, the application checks if an init file called '/etc/wgetrc/' exists; it does, and it is read. Next, we see that it tries to open a '.wgetrc' in my home directory, but this file does not exist, so this file is not opened.

The next example (Fig. 2, below right) shows that '/etc/resolv.conf' is being opened, and that a socket is opened to a DNS server, in order to resolve the address I asked for.

```
execve("/usr/bin/wget", ["wget", "-q", "http://www.google.com"], [/*
38 vars */]) = 0
...
stat64("/etc/wgetrc", {st_mode=S_IFREG|0644, st_size=4221, ...}) = 0
open("/etc/wgetrc", O_RDONLY|O_LARGEFILE) = 3
fstat64(3, {st_mode=S_IFREG|0644, st_size=4221, ...}) = 0
mmap2(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1,
0) = 0xb7ad2000
read(3, "###\n### Sample Wget initializati"..., 4096) = 4096
read(3, "on:\n#backup_converted = off\n\n# T"..., 4096) = 125
read(3, "", 4096)                           = 0
close(3)                                    = 0
...
stat64("/home/edb/.wgetrc", 0xbfe57a48) = -1 ENOENT (No such file or
directory)
...
```

Fig. 1

```
stat64("/etc/resolv.conf", {st_mode=S_IFREG|0644, st_size=88, ...}) = 0
socket(PF_INET, SOCK_DGRAM, IPPROTO_IP) = 4
connect(4, {sa_family=AF_INET, sin_port=htons(53),
sin_addr=inet_addr("195.130.131.5")}, 28) = 0
fcntl64(4, F_GETFL)                         = 0x2 (flags O_RDWR)
fcntl64(4, F_SETFL, O_RDWR|O_NONBLOCK)  = 0
gettimeofday({1234091526, 549043}, NULL) = 0
poll([{fd=4, events=POLLOUT, revents=POLLOUT}], 1, 0) = 1
send(4, "\372\312\1\0\0\1\0\0\0\0\0\0\3www\6google\2be\0\0\1\0\1"...,
31, MSG_NOSIGNAL) = 31
poll([{fd=4, events=POLLIN, revents=POLLIN}], 1, 5000) = 1
ioctl(4, FIONREAD, [367])                   = 0
recvfrom(4,
"\372\312\201\200\0\1\0\6\0\7\0\7\3www\6google\2be\0\0\1"..., 1024, 0,
{sa_family=AF_INET, sin_port=htons(53),
 sin_addr=inet_addr("195.130.131.5")}, [16]) = 367
close(4)
```

Fig. 2

Now, isn't this nice? We examined the internals of an application, without taking a look at a single line of code; we immediately learned where it stored some config files, which ones didn't exist, and how it translated the DNS entry to an ip address. ltrace works in a similar fashion, but instead of tracing system calls, ltrace gives you an idea of which functions are called and which are located in a dynamically linked library, see Fig. 3 (right).

ldd tells us that wget makes use of libssl (secure connections), libpthread (for creating threads), libz (compression), and libc amongst others. Libc is, in essence, the base of your system. It implements generic C functions like printf(), malloc(), and free(), and it implements this logic, often mapping them on system calls (e.g., a printf() maps to a write()). Now ltrace will tell us where our application makes use of functions provided by such a library. So, if we examine the output of

```
edb@lapedb:~$ whereis wget
wget: /usr/bin/wget /usr/share/man/man1/wget.1.gz
edb@lapedb:~$ ldd /usr/bin/wget
    linux-gate.so.1 =>  (0xb7f12000)
    libdl.so.2 => /lib/tls/i686/cmov/libdl.so.2 (0xb7ed8000)
    librt.so.1 => /lib/tls/i686/cmov/librt.so.1 (0xb7ecf000)
    libssl.so.0.9.8 => /usr/lib/i686/cmov/libssl.so.0.9.8 (0xb7e88000)
    libcrypto.so.0.9.8 => /usr/lib/i686/cmov/libcrypto.so.0.9.8 (0xb7d3c000)
    libc.so.6 => /lib/tls/i686/cmov/libc.so.6 (0xb7bde000)
    /lib/ld-linux.so.2 (0xb7ef8000)
    libpthread.so.0 => /lib/tls/i686/cmov/libpthread.so.0 (0xb7bc5000)
    libz.so.1 => /usr/lib/libz.so.1 (0xb7baf000)
```

Fig. 3

```
ltrace wget -q
http://www.google.com
```

we can find this snippet (some spaces omitted):

```
strlen("www.google.com")  = 14
dcgettext(0, 0x8075c8a, 5,
0x804e66d, 0xbf8e1761) = 0x8075c8a
getaddrinfo("www.google.com",
NULL, 0xbf8e1780, 0xbf8e17b4) = 0
calloc(1, 20) = 0x909c1e0
malloc(96) = 0x909c1f8
freeaddrinfo(0x909c100) = <void>
```

This is the ltrace counterpart of the DNS resolution we examined in the strace output. All the network communication is hidden by a simple call to 'getaddrinfo()'.

I hope that you now can appreciate the value of strace and ltrace. Both allow you to examine a binary's

internals without any extra effort. The only difference is that they will run a bit slower than normal, and this allows you to learn what the application is doing, and where something goes wrong.

Valgrind

Valgrind can be installed by typing:

```
sudo apt-get install valgrind
```

It is a collection of tools that do some advanced checking of applications. See http://www.valgrind.org for more information on the available tools. In this article, I will focus only on the most-used tool, called 'memcheck'. This tool will override libc calls that deal with

handling memory. And it will do some bookkeeping – is all memory (that is dynamically allocated) given back to the system, and is all the allocated memory still reachable?

Take a look at Listing 1 (below). This is some bad code. It calls a function leak() (lines 3 to 7) 10 times, which allocates 10 bytes and does not free it. Then it allocates some memory in the main function, and then goes into an eternal loop. The first thing I want you to do is to run this code, replace the for loop with a while(1) loop, and replace the malloc(10) with malloc(1000). Start the application, and see what happens with your system. Your physical memory will be filled, then your swapspace will be filled, and eventually the oom_killer (out of memory killer) will kill the fat process. Things like this are devastating to a system and its performance. You have just observed the effect of a memory leak. This is the painful part of dynamically asking for memory - it should all be given back! This is the example of a memory leak on an extreme timescale. Some applications, which leak only a couple of bytes an hour, might run perfectly for years - before all hell breaks loose. This is why valgrind is such a useful tool. This is the output of Listing 1 on my system after compiling it with

```
gcc -Wall -g leak.c -o memleak
```

displays Fig.4 (following page).

When I terminate the while(1) loop by pressing Ctrl+C, it tells me how many calls to malloc() I did, how much memory I allocated, and how much I gave back. And then it concludes that I have lost 100 bytes of memory in 10 blocks. This means that I asked for some memory that I can't reach anymore because I don't have a pointer to it ("definitely lost" in the output), and it says that I have also allocated 15 bytes in 1 block, which, at the moment of termination, I could still free because I have a pointer to it. This is why I had to write the while(1) loop. If I hadn't done that, valgrind would have concluded that I lost 115 bytes

```
01. #include <stdio.h>
02. #include <stdlib.h>
03. void leak()
04. {
05.     char * ptr = malloc(10);
06.     printf("malloc(10) points to: %p\n",ptr);
07. }
08. int main()
09. {
10.     int i=0;
11.     for(i=0;i<10;i++)
12.     {
13.         leak();
14.     }
15.     char * ptr = malloc(15);
16.     printf("malloc(15) in main: %p\n",ptr);
17.     while(1){}
18.     return 0;
19.}

Listing 1: leak.c
```

```
edb@lapedb:~/fullcircle/c-6$ valgrind --leak-check=full --show-
reachable=yes ./memleak
==7257== Memcheck, a memory error detector.
==7257== Copyright (C) 2002-2007, and GNU GPL'd, by Julian
Seward et al.
==7257== Using LibVEX rev 1854, a library for dynamic binary
translation.
==7257== Copyright (C) 2004-2007, and GNU GPL'd, by OpenWorks
LLP.
==7257== Using valgrind-3.3.1-Debian, a dynamic binary
instrumentation framework.
==7257== Copyright (C) 2000-2007, and GNU GPL'd, by Julian
Seward et al.
==7257== For more details, rerun with: -v
==7257==
malloc(10) now points to: 0x41a2028
malloc(10) now points to: 0x41a2068
malloc(10) now points to: 0x41a20a8
malloc(10) now points to: 0x41a20e8
malloc(10) now points to: 0x41a2128
malloc(10) now points to: 0x41a2168
malloc(10) now points to: 0x41a21a8
malloc(10) now points to: 0x41a21e8
malloc(10) now points to: 0x41a2228
malloc(10) now points to: 0x41a2268
malloc(15) in main: 0x41a22a8
^C==7257==
==7257== ERROR SUMMARY: 0 errors from 0 contexts (suppressed:
11 from 1)
==7257== malloc/free: in use at exit: 115 bytes in 11 blocks.
==7257== malloc/free: 11 allocs, 0 frees, 115 bytes allocated.
==7257== For counts of detected errors, rerun with: -v
==7257== searching for pointers to 11 not-freed blocks.
==7257== checked 52,132 bytes.
==7257==
==7257== 15 bytes in 1 blocks are still reachable in loss
record 1 of 2
==7257==    at 0x4025D2E: malloc (vg_replace_malloc.c:207)
==7257==    by 0x8048459: main (memleak.c:15)
==7257==
==7257== 100 bytes in 10 blocks are definitely lost in loss
record 2 of 2
==7257==    at 0x4025D2E: malloc (vg_replace_malloc.c:207)
==7257==    by 0x8048405: leak (memleak.c:5)
==7257==    by 0x8048443: main (memleak.c:13)
==7257==
==7257== LEAK SUMMARY:
==7257==    definitely lost: 100 bytes in 10 blocks.
==7257==      possibly lost: 0 bytes in 0 blocks.
==7257==    still reachable: 15 bytes in 1 blocks.
==7257==         suppressed: 0 bytes in 0 blocks.
```

Fig. 4

in 11 blocks (verify this!), because valgrind accounts for what really happens; it doesn't look forward to what might happen in the system.

One special thing I should still mention here: I told you earlier I compiled the code with the '-g' flag, which adds debugging symbols to the binary. This is why valgrind is able to tell us at which file and on which line number the error occurred. If I would compile the binary with

```
gcc -Wall leak.c -o memleak
```

the output would look like:

```
==7339== 100 bytes in 10 blocks are definitely lost
in loss record 2 of 2
==7339==    at 0x4025D2E: malloc
(vg_replace_malloc.c:207)
==7339==    by 0x8048405: leak (in
/home/edb/fullcircle/c-6/memleak)
==7339==    by 0x8048443: main (in
/home/edb/fullcircle/c-6/memleak)
```

It still is able to tell us we lost some memory, but it can no longer tell us on which file or line things went wrong.

So, the good news is valgrind telling us if we have a memory leak or not. The bad news is that we need to have a binary with debugging symbols in it if we wish to know the exact location of the leak. We can then recompile the executable to do some troubleshooting - for which we also need the sources!

Conclusions

In this article, I introduced some tools that allow you to do some easy troubleshooting and analysis on binaries,

without the need to have any sources or knowledge of these binaries. Next time, we'll even attempt to go a bit deeper, and then we'll take a look at using a real debugger.

Exercises:

• vmstat is a tool which reports system usage statistics, use strace to figure out which /proc/file(s) it uses to generate its output.

• Repeat the ltrace/strace

example on wget but now with an invalid URL, on which output is it easier to diagnose that the output was a faulty DNS entry.

• Read the strace manpage; will strace automatically follow child processes? What measures do you have to take when tracing a multithreaded application?

• Does valgrind automatically trace child processes?

• Try running valgrind on some of your favorite command line tools. And check if they behave properly

with respect to memory management.

• Which others tools are part of the valgrind suite, and how could these help you in writing higher quality applications?

Elie De Brauwer is a Belgian Linux fanatic, and apart from spending time with his family, he enjoys playing with technology, and spends his days waiting for Blizzard to finally release Diablo III.

APPLICABLE TO:

ubuntu kubuntu xubuntu

CATEGORIES:

Dev Graphics Internet Multimedia System

DEVICES:

CD/DVD HardDrive USB Drive Laptop Wireless

Today I'll show you how to install a web server and database, immediately ready for web development.

We will be setting up what's called a LAMP server (LAMP standing for Linux, Apache, MySQL and PHP) to create a localhost and also get into a bit of PHP programming. What this means is that you can edit and create a website in PHP — right on your own computer — without having to purchase server space. Why would you want to do this? Well, many web developers — whether they work freelance or for an organization — need a server to develop their websites on. Quite often, they'll write code, upload it to a server, refresh the page in Firefox, and hope for the best. Some advanced text editors and IDEs allow you to edit files directly on the server, but no matter how you upload your files, you still suffer from slow file transfers, broken connections, and other failed uploads. Using a localhost skips this uploading step and allows you to edit your entire website, right on your PC.

If you know how to forward port 80 on your router, you can even make it accessible to your friends or family. Although you could start to host websites professionally using the method I will show you today, I assume that if you are reading this article, you are not ready to learn about the intense security practices involved in actually hosting websites. Keep that in mind for down the road. What I am showing you is extremely simple (thanks to Ubuntu), but also extremely powerful.

For Windows users and developers who want (or have) to create websites in Windows (possibly using some tools that aren't available in Ubuntu, or because work dictates you use Windows), you can with this method set up an Ubuntu LAMP server through a virtual machine (try VMWare Server or VirtualBox — there are many tutorials online) to get a secure and reliable Ubuntu server while using Windows! Basically you'd install Ubuntu Server 8.04.2 or 8.10 into a virtual machine, use the method I show you to setup a LAMP server, and then access it via the IP address in your browser.

Installing a LAMP server in Ubuntu

Installing a LAMP server in Ubuntu is extremely easy, and you can do it in one of two ways:

Method 1

```
sudo tasksel install lamp-
server
```

Note: This was introduced in Ubuntu 6.10 (Edgy Eft), and works up to and including 9.04 (Jaunty Jackalope — currently in development). For 6.06 (Dapper Drake), you simply need to install 'tasksel' before trying to run tasksel.

Now, what has this magical program done? Running the above command is the same as running:

```
sudo aptitude install apache2
apache2.2-common apache2-mpm-
prefork apache2-utils
libapache2-mod-php5 libapr1
libaprutil1 libdbd-mysql-perl
libdbi-perl
libmysqlclient15off libnet-
daemon-perl libplrpc-perl
libpq5 libwrap0 mysql-client-
5.0 mysql-common mysql-server
mysql-server-5.0 openssl-
```

```
blacklist php5-common php5-mysql
ssl-cert tcpd
```

Which is easier to remember?

But, as you can see, this method installs all sorts of packages that you may not need nor want on your server (such as Perl, PostgreSQL, and SSL libraries). Now, are these packages bad for your system? Not at all. It depends on how lean you want your server to run, and how many packages you want to support. For now, "sudo tasksel install lamp-server" should suffice perfectly for the beginner or someone who needs a LAMP server up and running quickly.

Method 2

This method is similar to Method 1, with the exception of allowing you to select which packages you'd like to install.

For many, this would suffice to set up a LAMP server:

```
sudo aptitude install apache2
libapache2-mod-php5 mysql-server
php5
```

The problem with this is you're not exactly customizing your installation. Does everyone need customization?

Well no, but once you start using Wordpress, MediaWiki, Drupal, or any serious programming, you may need certain libraries installed. Don't worry, you'll learn which libraries are required as you go along.

So, suppose you are programming in PHP, and you require some image manipulation libraries, security libraries, or even the command-line interface (meaning you can run PHP through the terminal, not just through Firefox), then you need to customize your LAMP server installation.

Thus, for my LAMP server, I would install the following packages:

```
sudo aptitude install apache2
libapache2-mod-php5 mysql-
client-5.0 mysql-server-5.0
php5 php5-cli php5-curl php5-
gd php5-imagick php5-json
php5-mcrypt phpmyadmin php5-
mysql
```

Note my inclusion of 'phpmyadmin'. phpMyAdmin (as discussed in my first article) is a great tool for managing your

MySQL server. We won't go into databases today, but just remember how easy it is to install. To access it, simply point your browser to http://localhost/phpmyadmin and enter in 'root' as the username and the password you chose during the install.

Don't be afraid to customize, remix, and explore!

Testing your localhost

Once you successfully complete Method 1 or Method 2 for installing your LAMP server, open Firefox (Applications > Internet > Firefox Web Browser) and open:

http://localhost

There should simply be black text on a white background that says "It works!". If you don't see those words, hop on over to our sub-forum at Ubuntu Forums for help.

Now where do I put my files!?

Right now, Apache is reading your files in a (to you, probably obscure) directory called

/var/www/. There are two ways of changing this:

1. Delete /var/www/, and make a link/shortcut to a folder on your desktop or in your home directory, or:

2. Edit the Apache config file, and change all instances of /var/www/ to a folder on your desktop or in your home directory.

For 1, simply run this in the terminal:

```
sudo rm -r /var/www && mkdir
$HOME/Desktop/localhost && sudo
ln -s $HOME/Desktop/localhost/
/var/www
```

That line first removes /var/www/, then creates a folder at $HOME/Desktop/localhost (so, if my username was 'brett', it would create a folder at /home/brett/Desktop/localhost), and finally creates a shortcut from $HOME/Desktop/localhost to /var/www. This will successfully trick Apache into thinking the files are actually sitting in /var/www.

Note: if you want the folder to be somewhere other than your desktop, then simply edit the line above to

where you would like to store your development files (e.g. change $HOME/Desktop/localhost (both instances!) to $HOME/Documents/www (or whichever folder you'd like)).

For 2, edit the file "/etc/apache2/sites-available/default" by replacing "/var/www" with a folder of your choosing, and then reload Apache:

```
sudo /etc/init.d/apache2
reload
```

Whether using #1 or #2, check http://localhost in Firefox once more to make sure it is working. At this point, you should simply see a directory listing of the folder we just created (e.g. $HOME/Desktop/localhost/), which, more than likely, is empty and shows nothing.

Brett Alton is an Ubuntu enthusiast, computer technician and software engineer from Toronto, Canada

16

HOW-TO
Written by Ronnie Tucker

INSTALL CRUNCHEEE TO EEE PC

CrunchBang is an Ubuntu-based distribution. CrunchEEE is based on CrunchBang, but uses the OpenBox GUI and installs only necessary applications, mainly Internet ones, but with the Array.org kernel. The Array.org kernel is a modified Ubuntu kernel with all the necessary EEE patches installed, so it's a great time saver.

We'll install CrunchEEE from a bootable USB flash drive (USB stick). First, download the ISO image from crunchbanglinux.org. Next, see FCM#21 for an explanation of how to convert an ISO image to a bootable USB stick.

The first thing to do is insert the bootable USB stick into one of the EEE PC's USB ports. Next, turn on the EEE PC, and, as soon as you see the blue ASUS screen, hold down the ESC key. You'll now be presented with a terminal-style screen (below)with two boot options: boot from (1) the internal hard drive, or (2) the USB stick you have plugged in. Choose the USB stick.
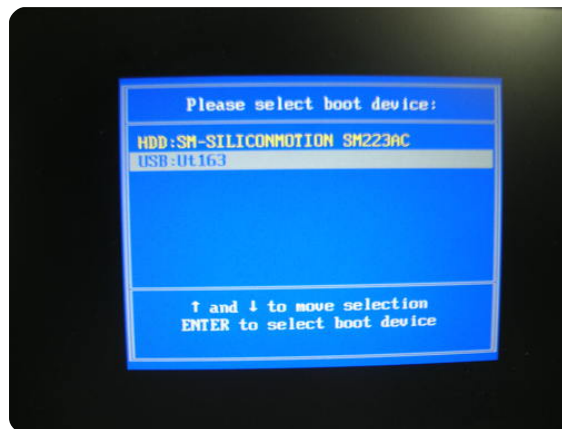
Sit back, wait, and watch, as your EEE PC boots CrunchEEE from the USB stick. Like most distros, it's a 'live' distro. You can try it out before you install it.
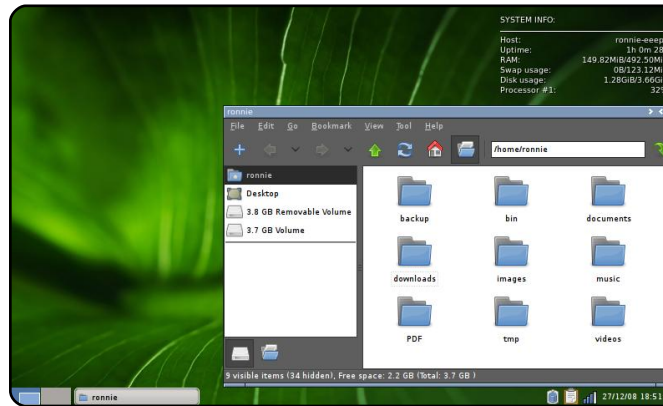
CrunchEEE opens with a black background, with Conky superimposed over the top right, giving basic information such as CPU percentage, memory used, and a list of hot-keys. Take the time to properly test-drive any distro before you install it to the hard drive. Make sure you at least test the sound and Wi-fi, since they are the main problem-areas in putting a distro on the EEE PC. To test the webcam, try Skype.

To begin installing CrunchEEE to the EEE PC hard drive, right click on the desktop and select 'install' from the main menu.

Installing CrunchEEE is the same as installing any other Ubuntu-based distro. Seven steps, and you're done. If you're

not sure of the steps to install an Ubuntu-based distro, check FCM#13 where we describe an Ubuntu installation step-by-step. What was nice about installing CrunchEEE was that the fonts had been made smaller, so the installation's windows fitted nicely to the EEE screen. No annoying ALT + move, or having to turn off Compiz to be allowed to ALT + move!

The default theme in CrunchEEE (above) is quite bland – it's a black background with Conky having white text, and the window decoration is black and grey. Don't worry, CrunchEEE comes with quite a few themes installed. A few clicks later, a change of wallpaper, and CrunchEEE is looking much nicer.


The steps outlined in creating the bootable USB stick (FCM#21), booting from it, and installing the distro (in this case CrunchEEE) are exactly the same for all Ubuntu-based distros, so feel free to download and try several distros. I've tried Kubuntu 8.10 (with KDE 4) and Xubuntu 8.10 (both of which need patches on the default installations to get some features working). I even tried Puppy Linux on my EEE before choosing CrunchEEE. That's one of the great things about Linux: choice!

Ronnie Tucker is Editor of Full Circle magazine, a recent GNOME convert, and artist whose gallery of work can be seen at www.RonnieTucker.co.uk.
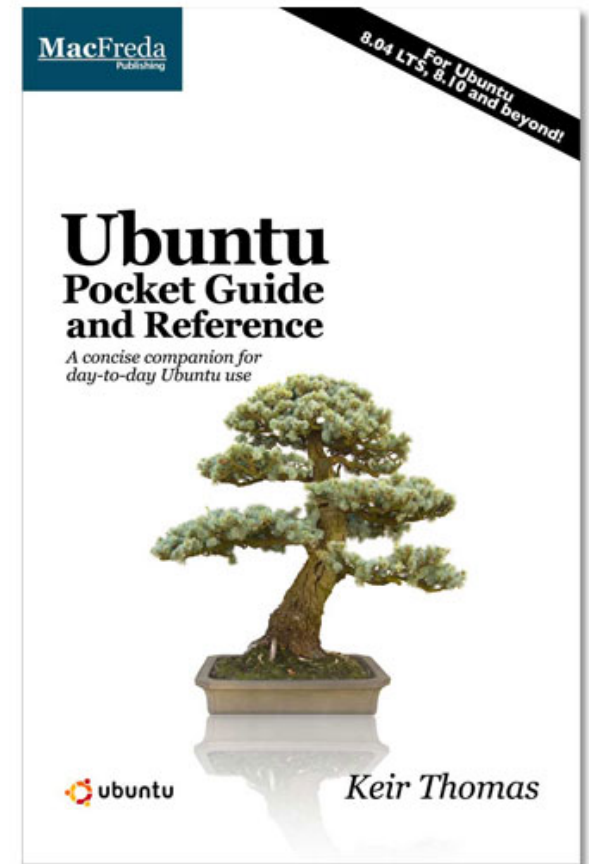
$9.94 from Amazon.com
or
FREE from

www.ubuntupocketguide.com

18
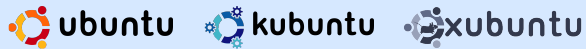
2008 was my year of Linux on the Desktop, as it was for many of my relatives, friends, and colleagues (see http://www.p-s.co.nz/wordpress/). But that doesn't mean Ubuntu is right for everyone. Here are some thoughts on how to proceed with spreading Ubuntu, one machine at a time.

Make Your System Attractive

Even if you don't see the value of an aesthetically pleasing desktop, most potential users of Ubuntu will. Being able to drag-rotate the desktop cube around is always a good selling point! Here is my dual-screen desktop. The theme is Dust, and it uses custom wide-screen wallpaper I made using Inkscape.

Choose Install Option

Firstly, determine what sort of installation will suit the user. A little thought up-front can save a lot of trouble later. And you'll need an ISO image file for Ubuntu – either on CD or USB. Grab your ISO from http://www.ubuntu.com/getubuntu/download. If you are going to set up multiple boot options, you may also want Gparted – http://gparted.sourceforge.net/download.php.

Dual Booting

This is the safest approach if the user has some Windows programs that you cannot get to work under WINE and if virtualization is not an option. The user will be able to read and write to all files in the Windows partition from Ubuntu. This can help with trouble shooting, and with a transition from one system to the other.

Dual-booting is particularly good for family machines because children gain access to all the games (er, educational software) from both worlds.

Pure-Play Ubuntu

This approach is the simplest to install, but make sure you won't need Windows again because dual-booting is more tricky to set up if you do it Ubuntu first/Windows second. And make sure to back everything up.

Virtualised XP

This option is excellent for newer machines, and is reasonably easy to set up using VirtualBox. You can run programs like MS Access, MS SQL Server, etc. inside – with no problems

(see image 2). And remember, you can share files between Ubuntu and XP only by creating shared folders.
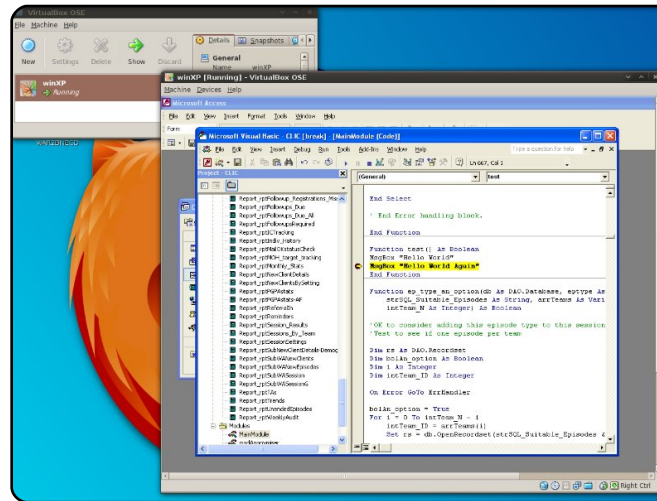
Who To Install Ubuntu For
(And Who to Leave Well Alone!)

Prime candidates:

• People who already know about Ubuntu – but want someone more experienced to set them up.

• XP users with infected machines – If they spend a lot of their time worrying about security and malware, then they might be open to Ubuntu. NOTE: If you end up having to repair a Windows machine, make sure the user lets you install Ubuntu as a dual-boot option. It helps with trouble shooting (e.g. you can read

the Windows partition from it) and, who knows, they might like it better. If you are expected to volunteer your time then it seems only fair.

• Vista users experiencing performance problems.

Avoid, or Approach with Caution:

• People who dislike using computers – They will probably dislike Ubuntu as well, and now they have someone to personally blame for their problems. Install Ubuntu only if they are expecting you to support Windows anyway.

• Business users in a highly Microsoft-oriented environment – There may be lots of applications that won't work under WINE, special network requirements, and so on. Dual-booting is an option, but a certain amount of caution is prudent. The OSS stack is very mature for some uses, but not for others. Be pragmatic.

• People With Essential Hardware That Won't Work In

Ubuntu. Test first.

### Eyes Wide Open
(What to Tell Them First)

OK, someone has asked you to install Ubuntu. What should you tell them before going ahead? Make the following points clear:

• It is their choice to install Ubuntu. You will do it for them, and will help them as much as possible later on, but ultimate responsibility for the result is theirs.

• Some hardware, and a lot of Windows-only software (e.g. from a lot of appliance store chains, etc.), may not work in Ubuntu. There is usually an alternative available for free within the open source world, but not always. If you are dual-booting, or you have XP virtualized, this may not be as important an issue.

Introduce New Users to the Community

Introduce new users to the resources of the Ubuntu world: First Circle; the Ubuntu forums (add bookmarks to their Firefox toolbar, etc.); other Ubuntu users; and Google of course. Plus, there are many excellent Ubuntu books out there now. Some resources suit some users better than others, but make sure they are aware of what they can do to help themselves.
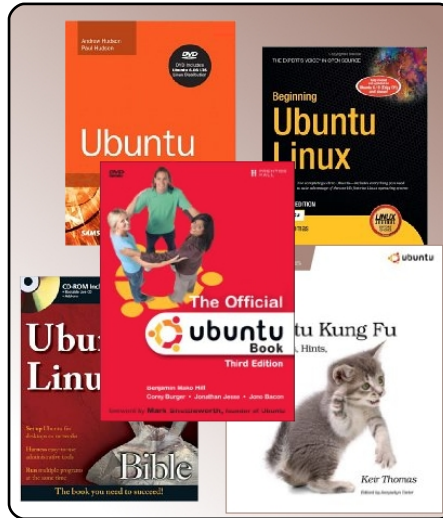
In my next article, I will look at "selling" Ubuntu to potential users.

Grant Paton-Simpson is a software developer from Auckland, New Zealand, with a PhD in Sociology. Married to Elizabeth with 4 lively children, and a custom database business (http://www.p-s.co.nz).

The Fridge
News for Human Beings

The Fridge is an information hub for the Ubuntu community, bringing together news, grassroots marketing, advocacy, team collaboration, and great original content.

Just like the family fridge at home, this is where we - the Ubuntu family - can put our best work on display for everyone to see. Whether you're working on advocacy and local marketing in a LoCo team, creating wonderful new worlds of Ubuntu in a derivative team, or building the freedom and technology of the future in a development team, we want to help you tell the community about your success.

http://fridge.ubuntu.com/

I have played with various distributions of Linux for the past five years. I would dabble in Red Hat running a web server, install Mandriva (Mandrake at the time) in a dual boot with XP, and actually built a kiosk for a tech school in my area using Suse Linux. I have also run various versions on Virtual Machines over Windows many times. I have always had a love/hate relationship with Linux. When I could get things working it was great! However, when I just needed something such as my sound card to work, I would find that 2 hours of compiling a driver just wasn't worth it.

I have never been able to switch to Linux as my main OS because of some of the issues that go along with it. Last week, I decided to give it another go. No dual-boot system this time either, I thought. Any time I have tried that in the past, I would always end up booting the Windows

partition out of pure geek laziness. There is a plethora of Distros available now, but I decided to go with Ubuntu. I guess I did this because of the tremendous amount of online support available. Also, The Geek has a complete section of useful Ubuntu and Linux tips!

I went with the default Gnome GUI at first. I was able to easily maneuver my way through Ubuntu pretty well. When it came to installing drivers for my Nvidia card, it was an easy and painless process. I also have a Creative X-Fi sound card - but I was not so lucky with this. There is a beta driver for the X-Fi card from Creative Open Source, but the bad thing for my set-up is that it does not currently support 5.1 surround sound. I am sticking to my integrated sound and will let my X-Fi card sit and collect dust for now.

I was really enjoying Gnome and my new Linux environment, when I decided to allow myself the

> " KDE … is a great choice if you are ditching Windows…

experience of KDE. Installing KDE with the default Gnome desktop environment is easy enough - just follow The Geek's Guide [1]. I have to admit that the KDE environment is a great choice if you are ditching Windows. KDE offers a lot of functionality right out of the box. With the release of KDE 4.0, it gets even better - a beautiful interface, familiar start menu, the ability to browse directories as root with just a click, and a lot more. I am now a Kubuntu fan, and I intend to do a clean install of the Kubuntu 64 system.

[1] The Geek - http://www.howtogeek.com/howto/ubuntu/install-kde-kubuntu-on-ubuntu/

Grant, Rickford

360 pages
June 2008

ISBN:
9781593271800

No Starch Press

Ubuntu For Non-Geeks covers every angle of Ubuntu - from trying it and installing it to customizing it. But where this book differs from others is in its project-based teaching.

In each chapter, Grant takes the user, step by step, through a small task. For example, in chapter five he teaches new users to use Synaptic by showing them how to install Frozen Bubble. Giving them Frozen Bubble will surely destroy their free time, but at least the user now has the knowledge and confidence to install, and uninstall, applications.

As previously mentioned, the book covers trying Ubuntu, installing it, getting used to the desktop, internet applications, customizing the look and feel of your new install and, of course, the terminal. But Grant leaves the terminal discussions until at least half way through the book. By that time the user is probably quite confident using Ubuntu and quite happy to tinker with the terminal. After that, the user reads about how to set up and install printers and scanners, adding fonts, business, graphic, audio, video and DVD applications and finally security.

I'd have no hesitation recommending Ubuntu For Non-Geeks to a new user as it shows new users how to do things step by step, it doesn't just explain it in text and leave the user to make assumptions. This book is definitely the best one I've read, so far, aimed at the new user.

To win a copy of *Ubuntu For Non-Geeks 3rd Edition*, answer this simple question:

Which game is shown being installed in Ubuntu For Non-Geeks?

Email your answer to: competition@fullcirclemagazine.org.

Deadline is Sat. 21st March 2009. The winner will be announced in FCM#23.

http://nostarch.com

Many thanks to **No Starch Press** for supporting Full Circle with this review copy of *Ubuntu For Non-Geeks*

Behind MOTU is a site featuring interviews with those known as 'Masters of the Universe' (MOTU). They are the volunteer army of package maintainers who look after the Universe and Multiverse software repositories.

Age: 21
Location: Orvieto, Italy
IRC Nick: emgent

How long have you used Linux, and what was your first distro?

I started with slackware in 1999, version 4.0 if I remember well. After that I switched to Gentoo Linux for my home-based servers, and Debian for desktop, but – intrigued by Ubuntu Hoary

Hedgehog – I installed it after its release in April 2005. From that moment, Ubuntu was always on my box somewhere, sharing my time with Aurox fedora based distribution for some time that I developed.

How long have you been using Ubuntu?

Since April 2005.

When did you get involved with the MOTU team, and how?

I started to contribute to the MOTU team when I decided to stop my work in Gentoo and Aurox development. I created a Launchpad account on 2007-11-09, and I started my contributions to the Ubuntu Security Team with Security fixes to packages, auditing, penetration tests on Launchpad, Ubuntu website, and other stuff (and I found some nice security bugs in these, now all fixed).

What helped you to learn packaging, and how Ubuntu teams work?

About packaging - there are a lot of great resources. I used the DebianMaintainer guide, and the Ubuntu packaging guide; the Ubuntu School logs, and Debian Policy, have been good places to consult when I didn't know anything, or was unsure. Now, thanks to Daniel Holbach, we have Ubuntu videos too - an important asset to start.

For questions, there is a #ubuntu-motu channel on freenode, another very important resources to solve possible problems.

What's your favorite part of working with the MOTU?

I think that working with lots of people that have the same

24

interests, views and goals.

Any advice for people wanting to help out MOTU?

Please read http://www.ubuntu.com/community/participate, and join – we need help to build the best distribution, so that we can thus conquer the world!

What are you going to focus on in Intrepid?

I'm working mainly on Rapache (https://launchpad.net/rapache, available in Intrepid Universe) – a GUI for configuring and managing apache2. Rapache is written in python+GTK coming out of an idea after UDS Prague. In this last period, I have also taken the WindowMaker package (lightweight, loving desktop manager), that was perhaps a bit too abandoned, as I am a wmaker developer, my son heard that I was the first person responsible for it. In addition to this, I work as usual in auditing, penetration test, and fix the security vulnerabilities contained in packages.

**How do you think Intrepid will be special for our users?**

The best WindowMaker ever seen, for stability and speed. After that, I think that rapache will be a good tool for all sysadmins, will facilitate the sysadmin with their configurations of the web server, and I hope will be a good tool to convince some sysadmins who uses Microsoft Windows to switch to Ubuntu.

Favorite quote?

"If God is, man is a slave." – Mikhail Bakunin

What do you do in your other spare time?

I do political activism (not political parties), and I'm involved in movements formed by voluntary associations who are working for various things. I do these with my girlfriend.
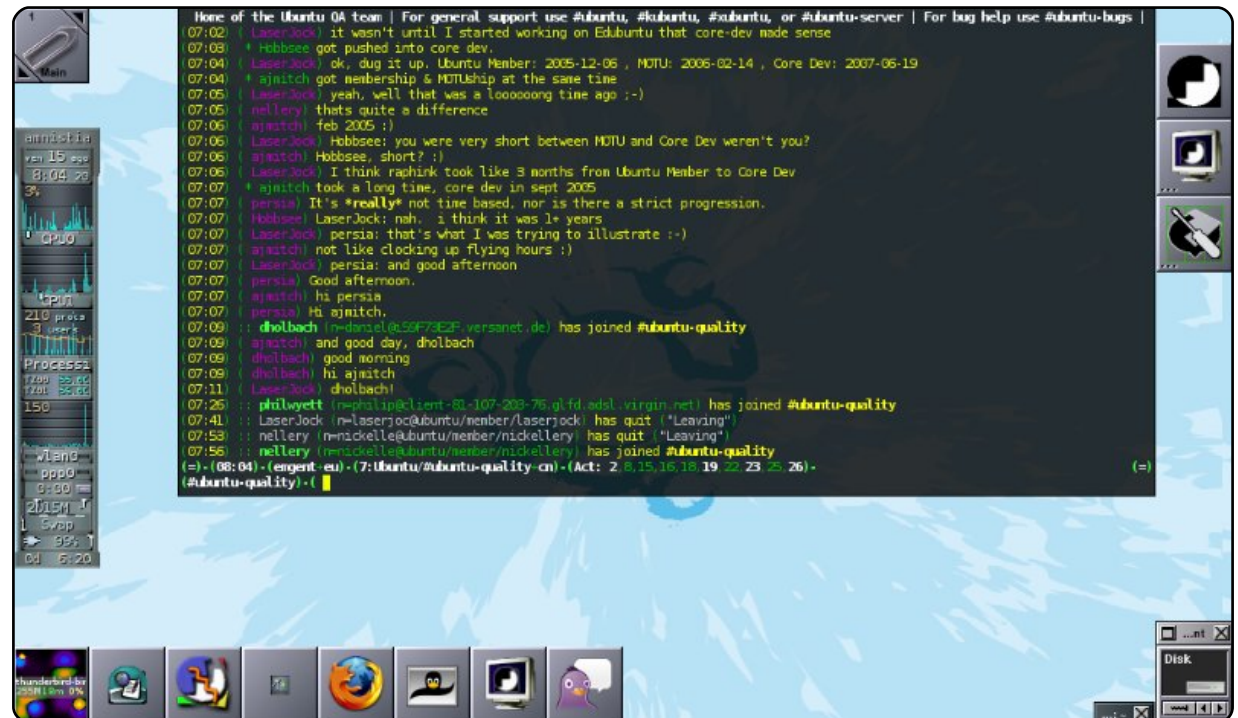
LETTERS

My husband and I were talking the other night, I have changed most of our six machines into Ubuntu and we now have two new grandchildren that are in Iowa, but we are in Pennsylvania. We need a way to keep in touch with our children, and grandchildren, and would prefer to do it via the Internet.

First we don't know what software we would need, we have powerful enough machines to run the software, camera and/or headsets, so that shouldn't be a problem. But then we need to know what hardware would work with the software (in Ubuntu) and of course which headset/phone we should look in to.

I was hoping maybe someone on your team, or a reader, would have useful information in this area that they could give to those of us who have family at long distances. Hopefully in the next few months, you might be able to

Letter Of The Month

Writer of Letter of the Month wins two metal Ubuntu case badges!

In the last issue of Full Circle, Mr Alsemgeest writes, "The only way to upgrade to EXT4 without losing data is to copy all your data to another partition or hard drive first". This is incorrect. Quoting [1]:

To convert an existing ext3 filesystem to use ext4, use the command

```
tune2fs -O
extents,uninit_bg,dir_index
/dev/DEV
```

Further, the answer to the Java/C#/popular languages question doesn't mention that C# (and other .Net languages) are widely used in the free software community thanks to the efforts of the Mono project.

Toby Smithe

[1] http://ext4.wiki.kernel.org/index.php/Ext4_Howto#Converting_an_ext3_filesystem_to_ext4

find someone who could make this happen?

Marianne Popp

Ed: *I've hardly touched VOIP (fancy abbreviation for telephone over Internet stuff), but Ekiga is installed by default on Ubuntu. For hardware, I'm really not sure. Readers, can you advise Marianne?*

I am a serious newbie and have gained some user ability with Ubuntu. My quest of the week is to try to understand Sane which is driving me INsane. I find it user unfriendly. It is not intuitive at all, at least not for me. So I was pleased to read in your mag about gscan2pdf and went in search of this scanning

elixir. Can it be substituted for Sane and its incantations? I am a long term Windows user and am using an HP all-in-one wonder device. The Windows software is easily understood without a manual to refer to. I would like to see gscan2pdf in the same light. Is it possible? I really would like to migrate to Linux.

Phil McCristle

Ed: *Xsane does seem somewhat daunting at first but all you need to do is click the 'Acquire' button (marked 1 on my diagram below) then 'Scan' (marked 2). Alter the number (marked 3) if you want to raise/lower the scanning detail (dpi). Normally 100 is enough, 300 is professional print quality*.
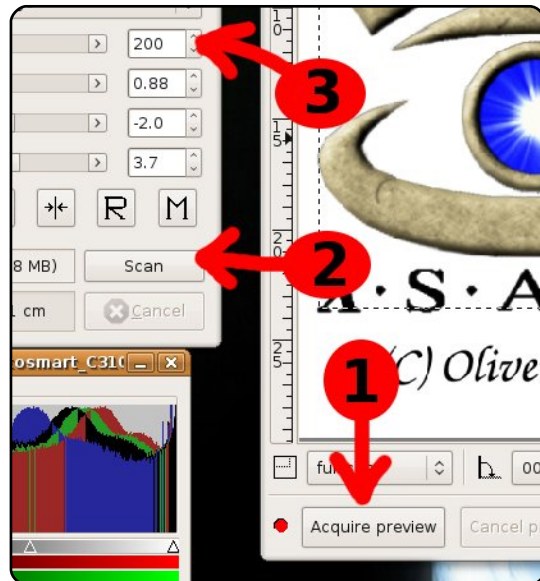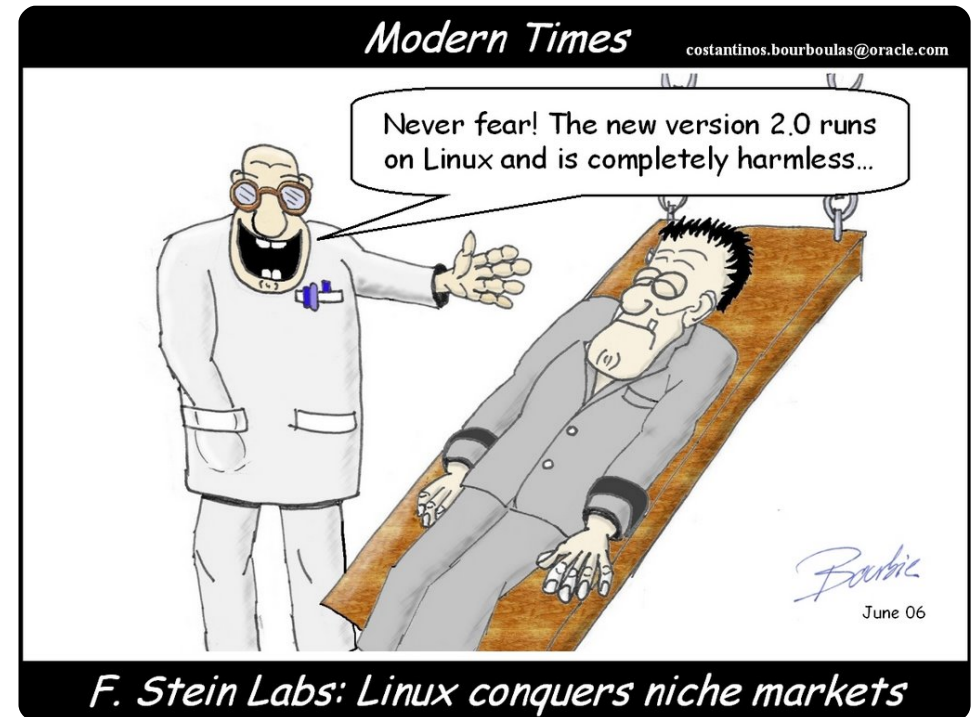
RE: Q&A of Full Circle Issue 21. The question asked if there was a newsgroup for Ubuntu. Your answer said it had closed. This is not completely true. Check: **alt.os.linux.ubuntu**. Unfortunately Google Groups does not carry the newsgroup, but it is available through regular news servers.

Lawrence Lucier & Eldergod

I downloaded and read some of your magazines today after seeing a link in the Ubuntu forums. Good stuff, but I wanted to respond to the two letters I read from reader "Vincent" about everyone using the word "Linux" to describe the computer operating system vs "GNU/Linux" which "might be a more appropriate (albeit cumbersome) name". I think that Linux has now come to be used as a generic term for Ubuntu, Red Hat, Suse, Slackware, et al the same as Unix has for all of its various incarnations. One has to admit that without Linus there would have been no Linux, just as without Stallman there would have been no GNU/Linux. The fact that Linus named his original software kernel 'Linux' is history. The fact that 'Linux' as a term has taken on a life of its own could not have been predicted.

GlenDobbs

Modern Times — costantinos.bourboulas@oracle.com

"Never fear! The new version 2.0 runs on Linux and is completely harmless..."

June 06

F. Stein Labs: Linux conquers niche markets

UBUNTU GAMES

Written by Edward Hewitt

GAME NEWS

• Much improved drivers from **nVidia** & **ATi** – the 180.22 drivers have just been released for nVidia cards. This driver allows more cards to work with Linux, improves performance, and supports OpenGL 3! ATi 9.1 driver also provides support for OpenGL 3.

• **World of Goo** (below) is highly rated for PC and the Wii. It has finally hit beta for Linux.
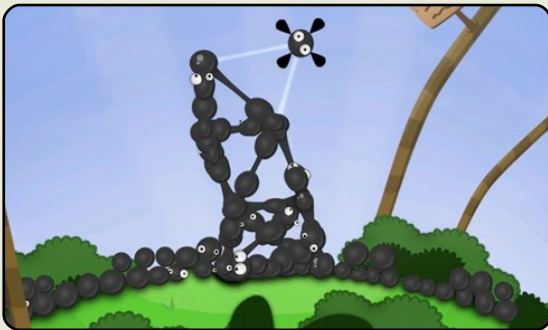
Linux is great! We all know that, but there are two major things I believe need resolving before we can say Linux is ready for the mainstream. The first is video editing and the second is gaming. Gaming is a very important part of the success of Linux, so there is much work to do before we can safely say that Linux is the platform of choice for gaming. In order to achieve this, I believe there are three key areas that need addressing: games, distribution, and services. Each of these areas is important for gamers, especially PC gamers.

The number and quality of games available for Linux is probably the most important area. Although it has many great games, it needs more games from major developers, such as EA or Activision. The big PC games need to be ported to Linux, such as World of Warcraft, Counter Strike, Battlefield, and Call of Duty. These are the games that all the gamers are playing. There is no point in trying to move gamers to Linux if

they can not play the games they love. Linux does have ways to make it easier for people to switch from Windows to Linux, thanks to great work from the Wine team. Games like Counter Strike do work in Linux. However, the support is not perfect. I much prefer native games to games played using Wine. It's important that we try to convince developers to develop games for Linux, and to port games to Linux. Hopefully, as OpenGL gets better, it will be easier for developers to do this. We may be able to help, by making the transfer from DirectX to OpenGL as easy as possible.

So, you may be thinking, now that we have these major games for Linux, how will we get our hands on them? The best

solution for Linux is digital distribution. I do not see Linux games ever being in shops. It would be far better to have a website or online store from which we can download Linux games. I believe a service like Steam is the best solution to getting games onto Linux. This solution is looking likely to happen, since Steam is apparently coming to Linux!

The final area is the services. There are many services which PC gamers use on Windows. These mainly consist of communication methods. The two major services are the gaming networks and the VOIP services. The gaming network is heavily used by gamers for talking to friends, joining games with each other, and for clans/guilds. The biggest network is Xfire, with over twelve-million users. It is important that this service is accessible for gamers on Linux. Luckily, there is a plugin for Pidgin, called Gfire (gfire.sf.net).

The second major network is fairly new: Steam. This online store now provides a social networking side for gaming, which is used in a similar way to Xfire. Sadly, there is no way to use the Steam community feature yet, unless you use Wine. The VOIP services are very important for clans/guilds to use during a match. The whole team needs to be in a VOIP channel to be able to talk about tactics during a match. The two main services are Teamspeak and Ventrilo. Teamspeak has a native Linux client, but Ventrilo does not. However,

Ventrilo does work very well in Wine.

If we are able to achieve success in all these areas, then we'll have a platform for Windows gamers to move to. I believe it is possible to achieve the last two areas, because we can develop these services within the Linux community. The difficult part is getting the developers on board. Apple is having a hard time trying to get the developers on board with their platform, so it is likely that Linux will have a bigger challenge.

Ed Hewitt, aka chewit, is a keen PC gamer and sometimes enjoys console gaming. He is also on the development team for the Gfire project (Xfire Plugin for Pidgin)

Q&A

Written by Tommy Alsemgeest

Q **I use an old Toshiba 4600 laptop to run Linux. The hardware supports a 1024x768 resolution, but, when installing any Linux distro, the screen shows only an 800x600 resolution - even in non-GUI sessions. I have installed and tested several distros, but they all react the same. My /etc/X11/xorg.conf doesn't have any entries like "800x600" or less, only entries for device and monitor. Is there any chance to have Linux use the full screen size?**

A To fix this you will have to edit your xorg.conf. This post on the Ubuntu Forums will tell you everything you need, even though it may seem outdated: http://ubuntuforums.org/showpost.php?p=129379&postcount=21

Q **I have just downloaded Ubuntu 8.10 and am trying to create a partition in my Windows Vista PC so that, when I boot up, I can choose to load Vista or Ubuntu. When I boot up with the CD in the drive, it does not give me an option to create Ubuntu within Windows. The options include: install, or run from disc. What option should I choose to install Ubuntu?**

A It is relatively simple to install Ubuntu inside Windows. You simply have to run the CD (from "My Computer"), and choose "Install inside Windows." This will bring up the Wubi installer, which will ask you some questions, then install Ubuntu onto a file inside Windows. Once it is finished, you can start into Ubuntu by restarting your computer and choosing Ubuntu.

Q **I would like to make the file 'firstprogram.py' executable. It is from a** Python tutorial but I did not see anything on how to make it executable.

A You make it executable by right-clicking the file, going to the permissions tab, and ticking the box "Allow executing file as program." You can also run the program from the terminal without making it executable by typing:

```
python firstprogram.py
```

Q **What type of OCR programs are out there for Ubuntu, and how do you load them?**

A This link should help you out, even if it was written for Ubuntu 7.04: http://www.howtoforge.com/ocr_with_tesseract_on_ubuntu704

MY DESKTOP

Your chance to show the world your desktop or PC. Email your screenshots and photos to: misc@fullcirclemagazine.org and include a brief paragraph about your desktop, your PC's specs and any other interesting tidbits about your setup.

I use Ubuntu 8.04. This system runs Compiz, Emerald, Cairo, and Screenlets. All sorts of things are tweaked and customized. The custom background was created in GIMP.

System specs:
AMD X2 5200,
BFG 8800 GTS,
2 Gigs of Corsair,
Two hard drives both 250 GB SATA.
One HD has XP (for gaming mostly)
and Ubuntu on the other HD.

Mike O'Donnell

I've been a Ubuntu user since version 5.10. I'm a PhD student in theoretical physics who uses Ubuntu exclusively for all my scientific tasks. It has never let me down, except at certain points (managing scientific papers, full-fledged PDF editing solutions), which I hope will be resolved soon.

This is my office desktop, with Intel Core 2 Duo, 2GB RAM, 500+160+160 GB hard-drives, 2.66 GHz, Intel Corporation 82G33/G31 Express Integrated Graphics Card. My computational work is primarily done on other parallel cluster machines, and hence I can spice up my desktop eye-candy with Compiz and also Cairo-dock. The wallpaper is from digitalblasphemy.com and my screen resolution is 1280x1024. I'm using Hardy Heron (Ubuntu 8.04) right now.

Vijay Kumar

31

I have been using Linux since 2005. I started off using YellowDog Linux on a Mac, then I switched to Ubuntu 5.04, and have upgraded through the years to, currently, Ubuntu 8.04. But now I am running it on a Dell Optiplex 270 machine, which has a 2.8 GHz Pentium 4HT, and 2 GB of RAM. I am dual-booting with Windows XP, which I hardly ever use. As for my desktop, I am using a background that I received from a friend of mine. I also use cairo-dock with the Mac OS X theme.

Jim Ricken

I've been using Ubuntu for about four years now, and I love everything about it. I love to customize things. I'm currently running Ubuntu 8.10 beta, and, so far, it's been great. Everything just seems to work, especially my USB Wi-Fi adapter, which I've had quite a bit of trouble with in the past. I got my wallpaper from gnome-look.org. It's called "wood-linux". The system stats on the right side is Conky - with a script I found somewhere, don't ask me where; the theme is NewHuman. My laptop is a Compaq Presario 2140. It's not much (AMD 2200+ with 512 MB RAM), but it works great, and I've had it for almost 6 years.

Justin

HandBrake

http://handbrake.fr/

While it was originally devloped for BeOS, Handbrake first gained prominence as an open source DVD ripper for OS X. However, in early 2006, a group of hackers forked HandBrake and renamed it Media Fork. One of the new features added was a CLI port to Linux. A year later, the projects merged, and now, HandBrake is one of the most prominent rippers on Linux, especially after the recent GTK+ GUI for the Linux port.

Unfortunately, it's not in the Ubuntu repositories yet; to install Handbrake, you'll need to use the official .deb at the download page.

## AcidRip
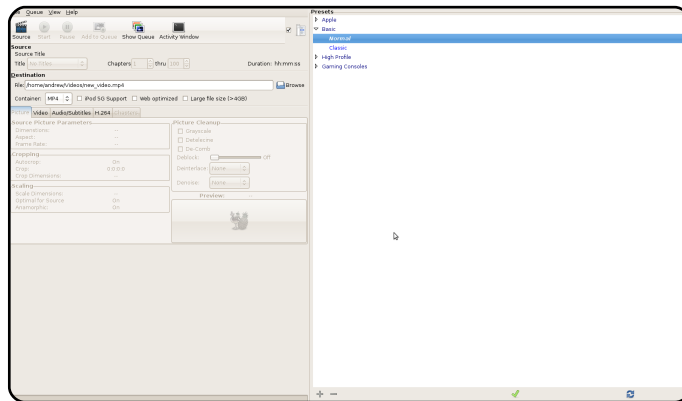
http://untrepid.com/acidrip/

AcidRip is a powerful, open source GTK frontend for MEncoder, the powerful but slightly confusing command line encoder. While it's focused on DVD-to-DivX conversion, it also outputs to MOV and (if you have the right components) MP3. It has an overwhelming number of features, since almost all the features of MEncoder are incorporated into AcidRip. It's certainly not the most user-friendly, but it's good if you like control.

To install AcidRip, use the '*acidrip*' package in the 'multiverse' repositories.
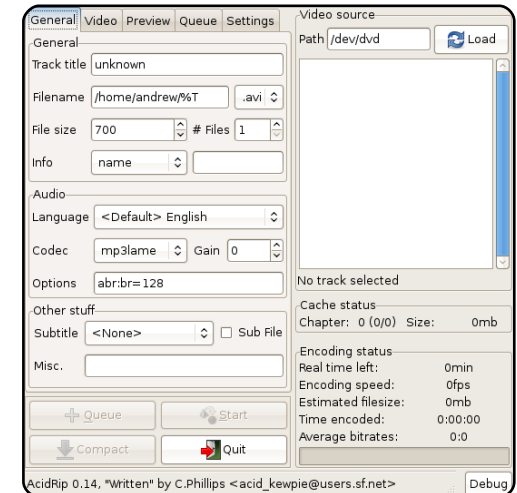
Thoggen
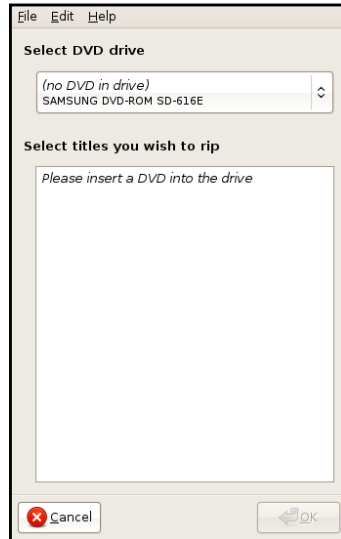
Thoggen is the simplest DVD ripper you'll ever find. You select the titles to rip, and click a button. Thoggen takes the titles, converts them to OGG, and then says bye bye. There are no advanced features (you can't even output to a format other than OGG), but that's what sets Thoggen apart. It's obviously not appealing to the uber-geek, but it does a great job of using the KISS principle to appeal to the less-technical crowd.

To install this "DVD backup" utility, use the `thoggen` package in the `universe` repositories.

dvd::rip

http://exit1.org/dvdrip/
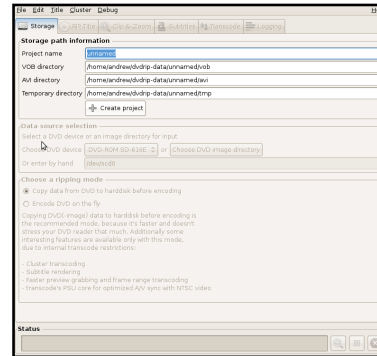
One of the oldest rippers around (it was started back in 2001), dvd::rip is the granddaddy of rippers. As it has aged, it has picked more and more features, from powerful projects to obscure tools like cluster control. It can output to a few different formats, but you'll forgive the lack of encoding diversity once you see the number of features it sports.
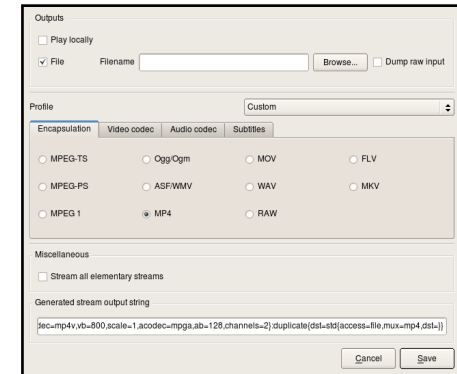
To install dvd::rip, use the `dvdrip` package in the `multiverse` repositories.

Andrew Min has been a Linux addict since he first installed openSuSE in VMWare. Learn more about him at http://www.andrewmin.com/

VLC

http://www.videolan.org/vlc/

VLC, the swiss army knife of media players, also plays and rips DVDs. While it's not the easiest to set up, there are multiple handy tutorials on the topic (http://url.fullcirclemagazine.org/24e316 is my personal favorite). And being VLC, it can encode to almost anything, including ASF, MPEG, Ogg, MOV, and even FLV (that's right... Flash video). Of course, it also has all the features you'd expect from a DVD ripper, plus a few special ones like messing with the codecs and subtitles.

To install VLC, use the `vlc` package in the `universe` repositories.

HOW TO CONTRIBUTE

We are always looking for new articles to include in Full Circle. For article guidelines, ideas, and for issue translation, please see our wiki:
http://wiki.ubuntu.com/UbuntuMagazine

Please email your articles to: articles@fullcirclemagazine.org

If you would like to submit news, email it to: news@fullcirclemagazine.org

Send your comments or Linux experiences to: letters@fullcirclemagazine.org

Hardware/software reviews should be sent to: reviews@fullcirclemagazine.org

Questions for Q&A should go to: questions@fullcirclemagazine.org

MyDesktop screens should be emailed to: misc@fullcirclemagazine.org

If you have questions, you can visit our forum: www.fullcirclemagazine.org

FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles and Full Circle is no exception. We need your Opinions, Desktops and Stories. We also need Reviews (games, apps & hardware), How-To articles (on any K/X/Ubuntu subject) and any questions, or suggestions, you may have.

Send them to: articles@fullcirclemagazine.org

Deadline for Issue #23:
Sunday 08th March 2009.

Release date for issue #23:
Friday 27th March 2009.